LETTERS TO THE EDITOR - I

Editor’s Note: The following exchange of letters contains two views of the role of end user. It was stimulated by Jim Sutter’s remarks at a meeting that the editor attended. Sutter is a very senior practitioner, having retired as CIO at North American Rockwell and is now a consultant. The editor asked Jim to write up his point of view and then asked Lorne Olfman, who was one of the team members of the SIM Advanced Practices Council project on Best Practices in End User Training to respond from the point of view of a researcher.

Additional comments on the subject of the role of end users are invited. The first two of these, by Jennex and Alter follow Sutter and Olfman. Accepted letters will be added to this exchange as they are received.

1. THE CASE OF/FOR THE MISSING USER

Jim Sutter
Peer Consulting Group

imsutter@aol.com

The Y2K remediation effort represented one of the largest IT undertakings ever. In terms of dollars spent, individual projects initiated, platforms impacted, geographic coverage, outside contractors employed, the scale and scope of “Y2K” tends to dwarf other Information Systems projects. Most IT departments carried off the task without a hitch.

This is quite surprising if one considers that the majority of IT projects launched during the last 20 years were regarded as outright failures; and those
that weren’t were either mild disappointments, or were over budget and/or missed deadlines. When survey after survey revealed the rather dismal track record of delivering IT projects on time, it was no surprise that nearly every organization in the public and private sector viewed the Y2K remediation problem as a major threat. No way would the planning be adequate, the resources be sufficient, the testing be thorough enough. Indeed, at the outset, it was difficult to get CEOs, mayors, governors, the media, and the general public to understand what the fuss was about. The requirement wasn’t even a requirement!

Given the history of project performance then, why has the effort yielded such optimistic forecasts, now that the conclusion of the effort is just days away? Why have tests in banking, transportation, financial services, and military readiness produced such positive results? Why are almost all large industrial companies feeling “quite comfortable” that their mission critical systems and processes are ready when just several months earlier their 10Ks and annual reports warned shareholders of major risks?

What is different about the Y2K effort? Could it have anything to do with Lack of User Involvement? As heretical as it sounds, a lesson we may take from the Y2K performance is that IT projects suffer many times from having “too many cooks in the kitchen”.

As the technology has become more pervasive, and as more and more end users become literate in terms of hardware, software, and telecommunications, the decision making process regarding new IT initiatives has tended to become more and more pluralistic. The problem is exacerbated by the myriad of choices ushered in during the so-called “client/server” generation of IT platforms. As a result, end user design sessions, intended to make the business rules and process requirements explicit, are made complex and even unnecessarily contentious by debates over platforms, vendors, products, and even release levels.
in the interest of gaining “ownership” and functional user “buy in”, IT project disciplines and accountabilities give way to major time consuming and schedule damaging debates over technical architectural components. Individual end users and various user groups or committees increasingly want to participate in decisions regarding technology decisions such as Windows 2000 migration, COM vs. CORBA, selection of web servers, etc. These debates, frequently encouraged by competing vendors, introduce time delays that lead to schedule elongation. Time is a very crucial element in managing technology-based projects. Any protracted delay almost always reduces benefits and increases the risk of success.

Excessive end user involvement can do more than just slow down the IT effort. It is unlikely that end users would have seen time series planning as the solution to excess inventory holding cost problems. Nor would end user groups have volunteered the changes and dislocations ushered in by the direct business-to-business electronic commerce model. Too often, senior management believed it discharged its governance responsibility over process design and systems definition by delegating the substance to various end user groups. In the end, these top-level managers have to decide how they want the business run if they are to keep up with modern entrepreneurs. Too many user committees blur the focus and unnecessarily expand the requirements.

The Y2K effort avoided both the time problem as well as the requirement issue. Top management did not have to delegate to user groups. It depended on the experience and expertise of the IT organization. By and large, the IT software issues were dealt with in a very straightforward manner.

While it can be argued that the Y2K issue required no new innovation to solve the problem, it remained a vast undertaking. And, it was, it most cases, completed ahead of schedule. One of the contributing factors of success may
have been the lack of complicating, time-consuming, excessive user involvement.

2. THE VALUE OR LEVEL OF USER PARTICIPATION – A RESEARCHER’S VIEW

Lorne Olfman,
School of Information Science
Claremont Graduate University
lorne.olfman@cgu.edu

Is there too much user participation in IS projects? It is possible, but without rigorous research in this domain, we are just guessing. Jim Sutter believes there is too much participation. He argues that users have become too involved in the systems development process and that the Y2K effort provides a good example of how effective IS shops could be without that level of “excessive” involvement. He notes based on anecdotal evidence, that most Y2K projects were completed early and under budget. Of course Y2K projects require essentially no innovation or definition of new requirements, and so need little user input. It is not surprising that costs would be lower.

Late, over budget development is not a new phenomenon. This rap on IS projects existed long before end user participation became conventional wisdom, and before end users were even allowed to provide input other than requirements. What Sutter does tell us is end users are becoming involved in decisions, such as platforms and products, that were previously the exclusive province of IS.

1 In 1989, Barki and Hartwick questioned the notion of user involvement as a useful construct. They argued that user involvement is an attitudinal construct, while user participation is a behavioral construct. Thus, in this letter, the term user participation is used to refer to the inclusion of end users in IS development projects. In 1994, Barki and Hartwick defined measurement instruments for these constructs.
As an educator of systems developers, I question the extent to which users should participate in projects. I believe that user participation should be limited to initiating project requests, giving and validating requirements and performing acceptance testing. The level of participation may vary depending on the type of project (large scale versus prototyping). I do not think that users should be involved in setting IS policies and strategic directions regarding platforms and the like. They should be involved in determining the type of applications and/or end-user tools that would run on a standard platform. Going to this level assumes a much different philosophy than standard arguments for user participation (e.g., Ives & Olson, 1984; Hirschheim, 1985).

The logic behind having users participate in the systems development process is the idea of getting requirements right. The persons whose requirements are being modeled and implemented in information systems are viewed as the ones who can best judge their own needs. Moreover, if they buy into the system, that is, if they are involved, then acceptance of the project is assumed to be more likely. Many IS academics argue that user participation early in the development process reduces total costs since correct requirements mean less time spent on redoing system code or performing extensive maintenance. However, users must also understand that time spent in defining requirements (whatever the level) increases project costs because all time spent that is not budgeted for translates directly into cost. Brooks (1995) showed that if a project is expected to be finished by a certain date, and it is behind schedule, no manner of additional resources can bring it back on schedule. In short, failure to budget adequately for user participation may be one of the causes of budget and schedule overruns.

As far back as 15 years ago, Ives and Olson (1984) concluded that research had not been able to demonstrate the value of (what they termed) end user involvement in systems development projects. They reviewed the then existing literature and found the evidence was at best neutral for the benefits of
user participation. They demonstrated that poor research methods and instruments were more pervasive than useful results. I am not aware of any further published empirical research that makes a conclusive case for the value of user participation. Sutter’s general concern about the dysfunctional nature of high levels of user participation cannot be refuted by empirical research results.

What may have led to potential excessive user participation? Perhaps systems development educators are not doing a good job of getting the message across, and IS practitioners let the pendulum swing too far to the other side. Or, possibly, as Sutter states, higher levels of user literacy and a wide range of choices contributed to IT initiatives becoming “more and more pluralistic.” Perhaps technology is changing too rapidly and users will never be satisfied. An alternative to Sutter’s hypothesis is that the success of Y2K projects is a matter of project teams performing a wide variety of maintenance tasks effectively because of the pressures brought to bear by Y2K fears.

Sutter’s questioning of user participation brings to light the need for researchers to focus on finding what levels of participation are most effective. Barki and Hartwick’s (1994) work on developing valid instruments for this research should be an impetus. Moreover, other key research on user participation should not be ignored. Hirschheim (1985) found that while there was no clear evidence as to the benefits of user participation in terms of project costs, there was a benefit in that users value the participation process. Newman and Noble (1990) showed that user participation is not a simple process. There are context-dependent situations that require different forms of participation. If developers use the wrong process in a particular situation, this would likely lead to poor results. Hirschheim and Newman (1991) argued that systems development is not “a normative process reflecting conventional economic rationality” (p. 29). Again, developers must understand the social nature of the development process to make it work effectively. Providing meaningful empirical studies that can help practitioners learn the key parameters for systems
development success will be a welcome addition to the IS discipline’s knowledge base.

REFERENCES


Having spent the last two and half years immersed in preparing the electric utility industry for Y2K, I read these letters with more than a casual interest. Since Y2K has now come and gone with no major problems we are being congratulated for a job well done. Also, many of us in Y2K are being asked was it all necessary. Sutter proposes that our Y2K efforts succeeded because of a lack of user involvement. If this is the case, then we now have a hugely successful economic case for never involving users again, but this is not correct. My experience is in the electric utility industry and I can only speak from that experience. However, that experience says that Y2K projects succeeded because of user involvement. Olfman's rebuttal suggests that we need to do research to determine the appropriate level of user involvement for determining project requirements. That is a good suggestion that should be done, but is not necessary for determining why Y2K projects succeeded. Y2K was not a requirements issue. Y2K is perhaps the first and only IS project where the requirements were fully understood: make software function properly in the year 2000. This statement may seem trivial but the point I wish to make is that to use Y2K as an argument for or against user involvement in requirements determination is inappropriate.

Y2K projects succeeded because of superior project management. As Sutter and Olfman point out, IS has a history of poor performance with respect to cost and schedule. This was the predominant reason a significant number of
Y2K projects in the electric utility industry were managed by users, not IS. Yes, IS had the responsibility for Y2K in almost all utilities, but the majority of these utilities recognized the risk posed by a fixed deadline and turned to the users for project management expertise. I am an example of this. Prior to Y2K I was an engineer at our nuclear facility who provided engineering and IS expertise. I applied for the Y2K team as a way of getting into the IS department since I was a recent Ph.D. graduate of an IS program. Yes, I have IS expertise, but that is not why I was brought in to help manage Y2K. I was brought in because I had expertise in managing time sensitive engineering projects: projects that had to meet the deadline. As Y2K became an industry effort with knowledge sharing being supported through the Electric Power Research Institute, it became apparent that the majority of project managers were like me, from the user organizations of our companies. In our discussions it became apparent that the reason so many users were involved were two fold. The first was that user organizations knew how to manage a project so that it would meet its deadline. The second was because there were users who understood the systems. Y2K was a success because projects were finished on time and because we did not overspend by fixing non-essential systems. The IS organization would have fixed everything, hence the huge scope referred to by Sutter. However, experienced project managers understood that the key to meeting a deadline is to manage the scope. Users were necessary to determine what systems had to be fixed and what systems could be overlooked. Establishing criticality of a system is not an IS function, only the users can determine what systems have the most impact on their job functions. Another user perspective was that not all date sensitive functions needed to be fixed. It has been stated that up to 45% of functions built into systems are never used. Hence the utility industry strove for Y2K Ready. This meant that a system would perform its critical functions but may not indicate proper time and date. Functions that were not critical or used were not fixed. Y2K verified that there were a significant number of not used or non-critical functions.
Y2K also verified that there were a significant number of programs not used but sitting out on the servers and platforms. One of the significant benefits arising out of Y2K was the retirement of a large number of systems that were no longer used. We also were able to eliminate a large number of multiple versions that were being supported. IS could not easily make this determination, user input was necessary to determine what code was used and what versions were no longer needed.

An issue not specifically mentioned by Sutter or Olfman is that of embedded systems. IS rarely supports these devices. However, the critical control systems relied upon by the electric power industry to produce and distribute power rely on the use of automatic, digital controls. Users had to be involved in this aspect of Y2K. One of the reasons for the initial high-risk estimates of Y2K was IS evaluations of risk in these devices. IS looked at what the devices were capable of doing and saw they needed time, hence they were risky. The users on the other hand understood how the devices were used and knew that in many cases these devices did not need time to perform their critical functions. Scope and cost management are again the issues. IS would have fixed the time and date stamping features of these devices. The users did not, as they pointed out that if you don’t use the date and time stamp and/or the device will perform its critical function regardless of what time or date the device is using, then why fix it. This reason is why the massive failures predicted by IS experts failed to materialize, dates just did not matter for the most part (yes, some devices are impacted by dates, but user/IS testing and experience found these out and they were repaired).

In conclusion, I suggest the success of Y2K projects was not an issue of user involvement in determining project requirements as stated by Sutter. Instead, success was due to the blending of user system and project management expertise with the coding and testing expertise of IS. I would also suggest that the perception of users is perhaps out of date. Olfman states that
users are becoming more knowledgeable with respect to IS and technology. While my level of IS training is unusual for users, a high level of training is not. Utility projects found a very high level of IS expertise among their users. I would be the first to say that the engineer-dominated electrical utility industry is not typical of business users and that this industry may be the exception rather than the rule. However, the blending of digital technology with traditional technology is creating a new class of user. This new user has a mix of IS and business expertise that may represent the evolutionary path of IS professionals. Perhaps the issue of user involvement is not one of should they be involved, but rather, which type of user should be involved and what roles should they fill as Olfman suggests. I issue the warning that our users are more capable than we give them credit and to try to exclude them from IS projects designed to provide them with tools is a recipe for IS organization disaster. If IS does not include the users then the users will get rid of IS and for better or worse do it themselves. There is anecdotal evidence to support this and Olfman is on the mark in calling for research in this area. The lesson of Y2K is not in user involvement but in how to manage projects. Y2K should be investigated for lessons in improving IS project management, a subject both Sutter and Olfman agree on.

January 19, 2000
4. TOO MUCH USER PARTICIPATION OR TOO MANY INCONSISTENT MEANINGS FOR “USER PARTICIPATION”?

Steven Alter
School of Business Administration
University of San Francisco
San Francisco, CA 94117, USA
alter@usfca.edu

This letter is a response to a November 1999 exchange of letters between Jim Sutter and Lorne Olfman in CAIS in entitled "The Case of/for the Missing User." [Sutter and Olfman, 1999]. Based on his experience as CIO of a Fortune 10 corporation, Sutter’s letter expresses an idea that sounds politically incorrect: Perhaps users should NOT be so involved in software development, especially when the issues involve topics about which users have little insight or knowledge, technical issues such as Windows 2000 migration, COM vs. CORBA, selection of web servers, and so on. He argues “IT projects suffer many times from having too many cooks in the kitchen." He compares the high success rate of Y2K remediation projects with the high rate of failure or disappointment with projects that had much more user involvement, especially when user sessions “are made complex and unnecessarily contentious by debates over platforms, vendors, products, and even in release levels” in the interests of “gaining ownership and functional user ‘buy-in’.”

When I first glanced at Sutter’s letter my gut reaction was disbelief since “anyone knows that user involvement is important and beneficial.” Then I took another look and realized that Sutter’s users were functional area managers and their representatives, people with enough clout to get involved in discussions of technical IT strategy whether or not they had much knowledge to contribute. These are people CIOs and high level IT managers view as “their users” but these aren’t the people I usually think of as users, namely, people who use
My first glance at Olfman’s response also raised questions. He starts “Is there too much user participation in IS projects? It is possible, but without rigorous research in this domain we are just guessing.” He notes that Ives and Olson [1984] concluded that research had not been able to demonstrate the value of (what they termed) end user involvement in systems development projects.” Later he says, “Sutter’s questioning of user participation brings to light the need for researchers to focus on finding what levels of participation are most effective. … Providing meaningful empirical studies that can help practitioners learn the key parameters for systems development success will be a welcome addition to the IS discipline’s knowledge base.”

My reaction, especially after re-reading Sutter’s letter, was that adding to the IS discipline’s knowledge base is NOT the issue in this particular case and that more research would NOT provide useful answers to Olfman’s very general opening question, “Is there too much user participation in IS projects?” The real issue is that this question is too broad. The users under consideration might have data entry roles, information usage roles, or management roles. Their form of involvement might be symbolic involvement, involvement by advice, involvement by weak control, involvement by doing, or involvement by strong control. [Ives and Olson, 1984] Usage of the information system might be voluntary or mandatory. The IS project might involve modifying technical infrastructure, fixing technical bugs such as the Y2K bug, building an information system from scratch, installing packaged software developed elsewhere, developing a prototype, developing a small information system through an end user computing approach, hooking into an industry supply chain, and so on. There are so many different types of users, different types of involvement, different usage situations, and different types of projects that broad generalizations about whether the entire world has too much user participation in IS projects are not useful.
Upon re-reading the Sutter/Olfman letters carefully I have come to agree with a lot of very sensible things they say. At the same time, the intended controversy about excessive user participation seems artificial. Sutter’s claim about too many cooks in the kitchen is about certain types of situations and is not about user participation in general. Similarly, his example of Y2K remediation is representative of technical IT projects in which users have little to contribute, but it certainly isn’t representative of most IS projects. Although the two letters raised many valid issues about effective user participation, the intended controversy is less a controversy than an easily resolved confusion about vocabulary and basic concepts. Sutter was clear enough about the situations he was referring to, but my personal associations and assumptions regarding IS/IT vocabulary and basic concepts initially misled me. In contrast with Sutter’s experience as a CIO, when I see the terms “user participation” and “IS project” together I typically do not think of business executives voicing uninformed opinions about inherently technical topics.

Ironically, this same type of issue about the need for a nuanced view rather than a broadbrush, one-size-fits-all, view in basic IS concepts was a major topic in another CAIS paper appearing same month as the Sutter/Olfman letters. This paper, “Dimensions of Information Systems Effectiveness,” [Seddon et al, 1999] built upon an earlier paper [DeLone and McLean, 1992] that attempted to summarize previous research about IS effectiveness. The new paper presents a framework for evaluating IS effectiveness based two variables, the stakeholder’s point of view and the type of system. It identifies five points of view for evaluating IS effectiveness (an uninvolved observer, an individual who wants to be better off, a group which wants to be better off, managers or owners, and a country or society that wants to be better off). It then identifies six different views of what system is being evaluated (an aspect of IT use, a single IT application, a type of IT application, all IT applications used by an organization, an aspect of a
system development methodology, and the IT function of an organization). The paper combines these two dimensions into a 5 X 6 matrix and argues that different measures of IS effectiveness are needed for each of the different combinations of stakeholder and system. The similarity with the relationship between user participation and IS projects is that one size does not fit all situations, whether or not the same terms, “user participation” and “IS projects,” happen to be used.

In November 1999, with around 30 years of IS research under our collective belts, these two CAIS articles raised issues about the definitions of basic IS concepts including user, user participation, and system success. I think this is a cause for concern about what Olfman’s last sentence refers to as “the IS discipline’s knowledge base.” At minimum, I think the range of different meanings and connotations associated with the most commonplace IS terms implies that we as a field need to pay more attention to basic concepts and how different people use them.

REFERENCES


http://cais.isworld.org/letters/2-1/

April 4, 2000

Copyright © 1999, by the Association for Information Systems. Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and full citation on the first page. Copyright for components of this work owned by others than the Association for Information Systems must be honored. Abstracting with credit is permitted. To copy otherwise, to republish, to post on servers, or to redistribute to lists requires prior specific permission and/or fee. Request permission to publish from: AIS Administrative Office, P.O. Box 2712 Atlanta, GA, 30301-2712 Attn: Reprints or via e-mail from ais@gsu.edu.